Adaptive Multi-Level Explicit Congestion Notification (AMECN)

Arjan Durresi, Mukundan Sridharan and Raj Jain

Presented by

Raj Jain

CTO
Nayr ty
San J 10-1277

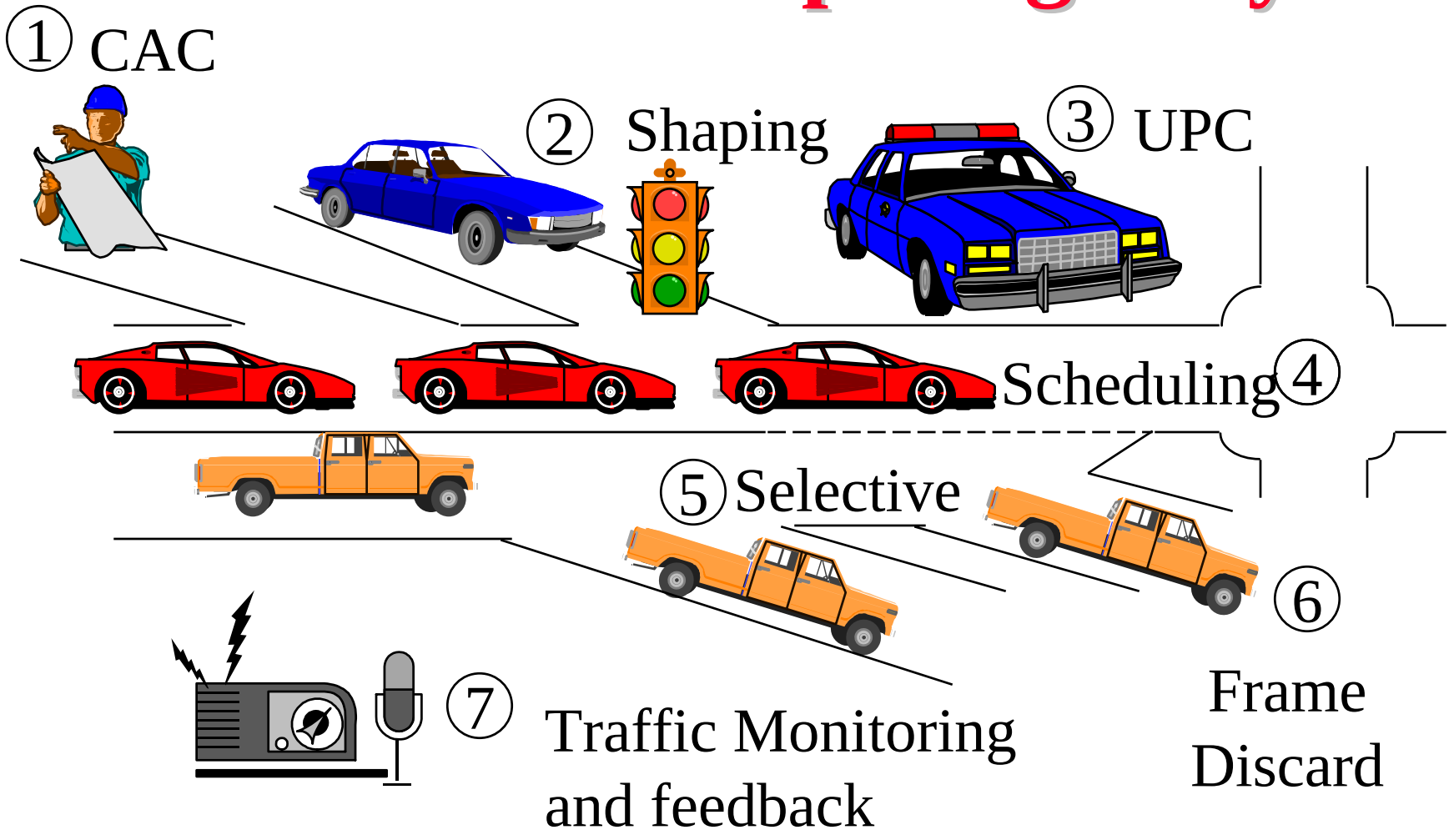
Raj Jain is now at
Washington University in Saint Louis
Jain@cse.wustl.edu
http://www.cse.wustl.edu/~jain/

tm

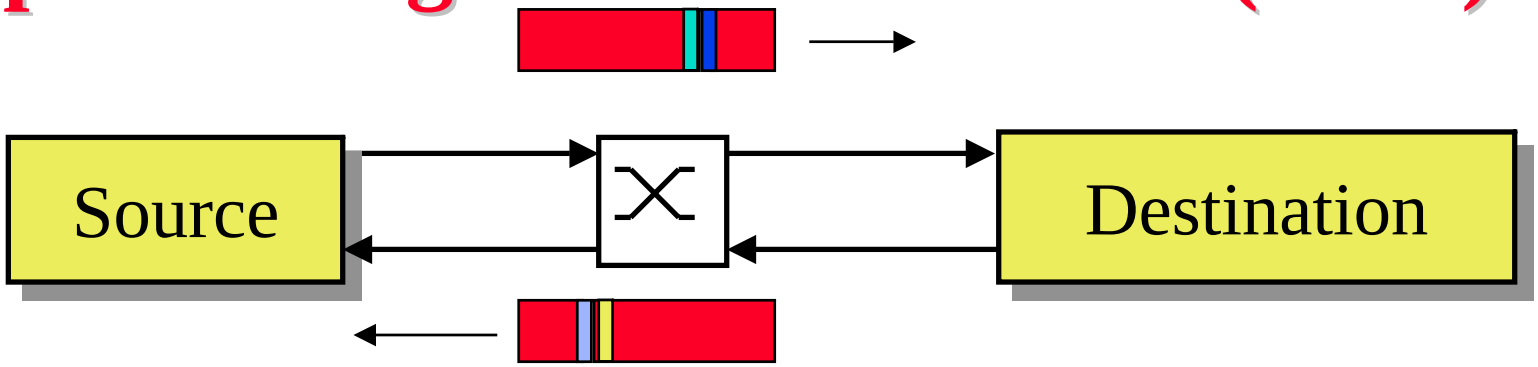
International Symposium on Performance Evaluation of Computer and Telecommunications Systems (SPECTS2004)
and Summer Computer Simulation Conference (SCSC 2004)
July 26, 2004

- Why Congestion management?
- RED, ECN and MECN
- AMECN
- Simulation results: ARED vs. AMECN

Traffic Management on the Information Superhighway

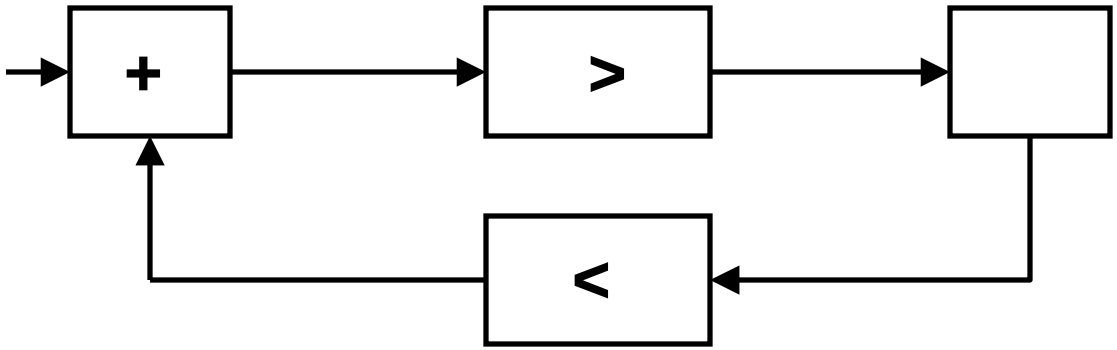


Explicit Congestion Notification (ECN)



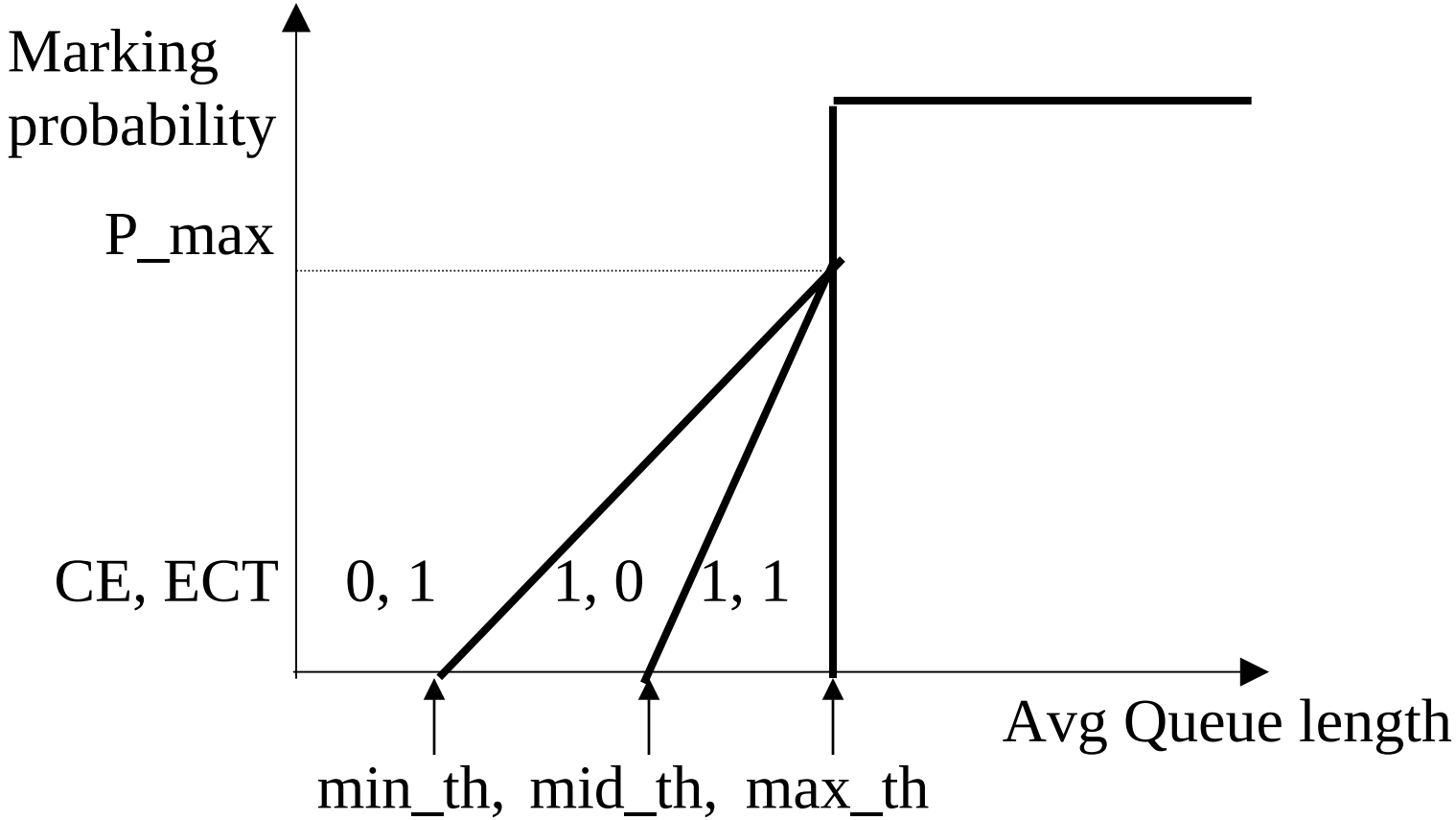
- Explicit Congestion Notification (RFC 2481 standard)
- Two bits in IP header:
  - Congestion Experienced (CE)
  - ECN Capable Transport (ECT)
  - Bits 6 and 7 in the ToS octet in IPv4, or the Traffic Class octet in IPv6
- Two bits in TCP header:
  - ECN Echo (ECE)
  - Congestion Window Reduced (CWR)

Multilevel ECN



- ECN – feedback control mechanism
- More feedback information
  ⇒ Enables more appropriate response from the source
- Better congestion control:
  - More effective use of network resources
  - Less losses
  - Less delay
- Use control theory to set parameters in RED, ECN and MECN

Router Marking



Router Response to Congestion: Marking CE and ECT Bits

CE bit	ECT bit	Congestion State
0	1	No Congestion
1	0	Incipient congestion
1	1	Moderate congestion
Packet drop		Severe congestion

Receiver Marking of CWR and ECE Bits

CWR bit	ECE bit	Congestion
0	0	No Congestion or non-ECN capable
0	1	Incipient congestion
1	1	Moderate congestion

TCP Source Response

Congestion State	cwnd change
No congestion	Increase ‘cwnd’ additively
Incipient congestion	Decrease multiplicatively by β_1
Moderate congestion	Decrease multiplicatively by β_2
Severe congestion	Decrease multiplicatively by β_3

Adaptive MECN

- The objective is to control the delay in each router by maintaining the queue near a target value: *target_queue*
- *P*_max is adapted to keep the average queue size with a target range half way between min_th and max_th.
- *P*_max is adapted slowly, over time scales greater than a typical round-trip time and in small steps. The time scale is generally 5-10 times the typical round-trip time of the network.
- *P*_max is constrained to remain with the range of [0.01, 0.5]
- Instead of multiplicatively increasing and decreasing *P*_max, we use an additive-increase multiplicative-decrease (AIMD) policy.

The AMECN Algorithm

Every *interval* (0.5) seconds :

if ($avg > target$ and $P_{max} <= 0.5$)

increase P_{max}:

$\alpha = 0.25 * \frac{avg-target}{target} * P_{max}$;

$P_{max} = P_{max} + \alpha$;

elseif ($avg < target$ and $P_{max} >= 0.01$)

decrease P_{max}:

$X = 0.17 * \frac{target}{target-min}$;

$\beta = 1 - X * \frac{target-ave}{target}$;

$P_{max} = P_{max} * \beta$;

Variables:

avg: average queue size

Fixed parameters:

$interval$: time; 0.5 seconds

$target$: target for avg;

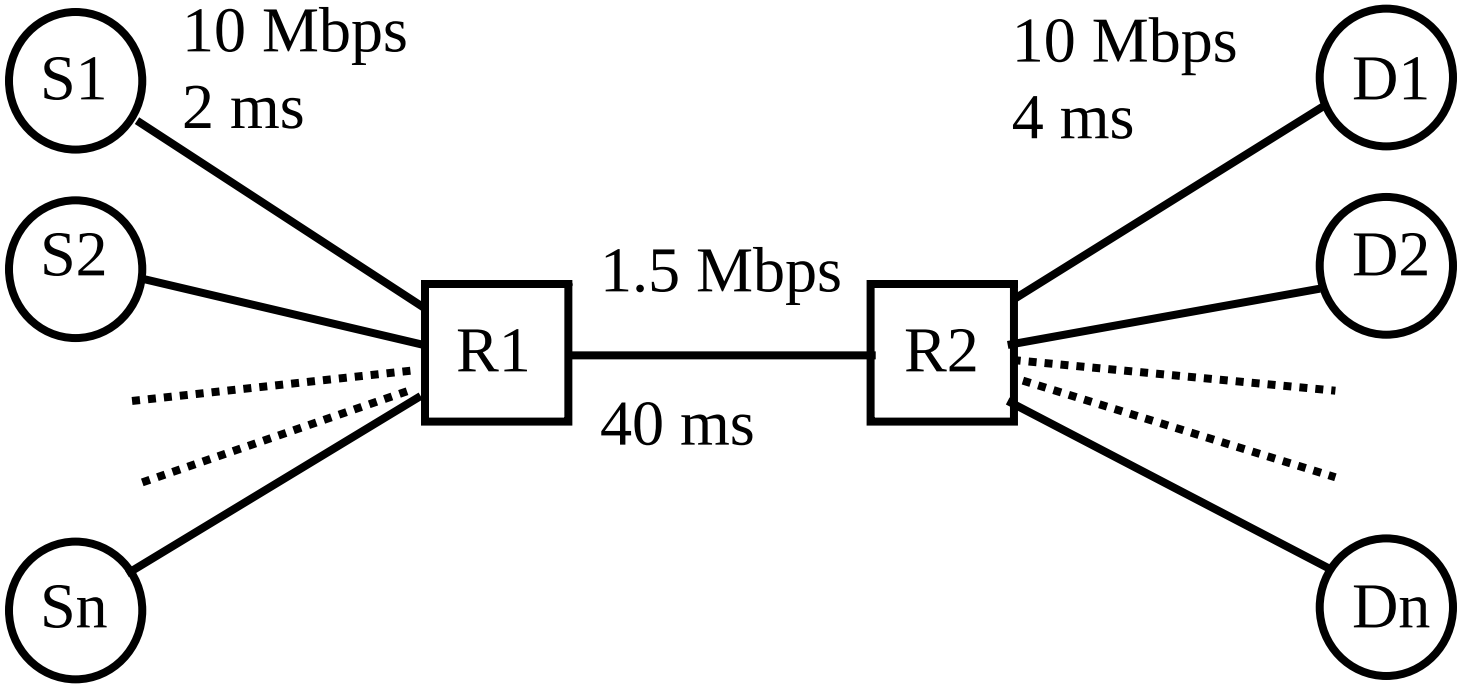
$[min_th + 0.4 * (max_th - min_th), min_th + 0.6 * (max_th - min_th)]$

α: increment; $0.25 * \frac{avg-target}{target} * P_{max}$

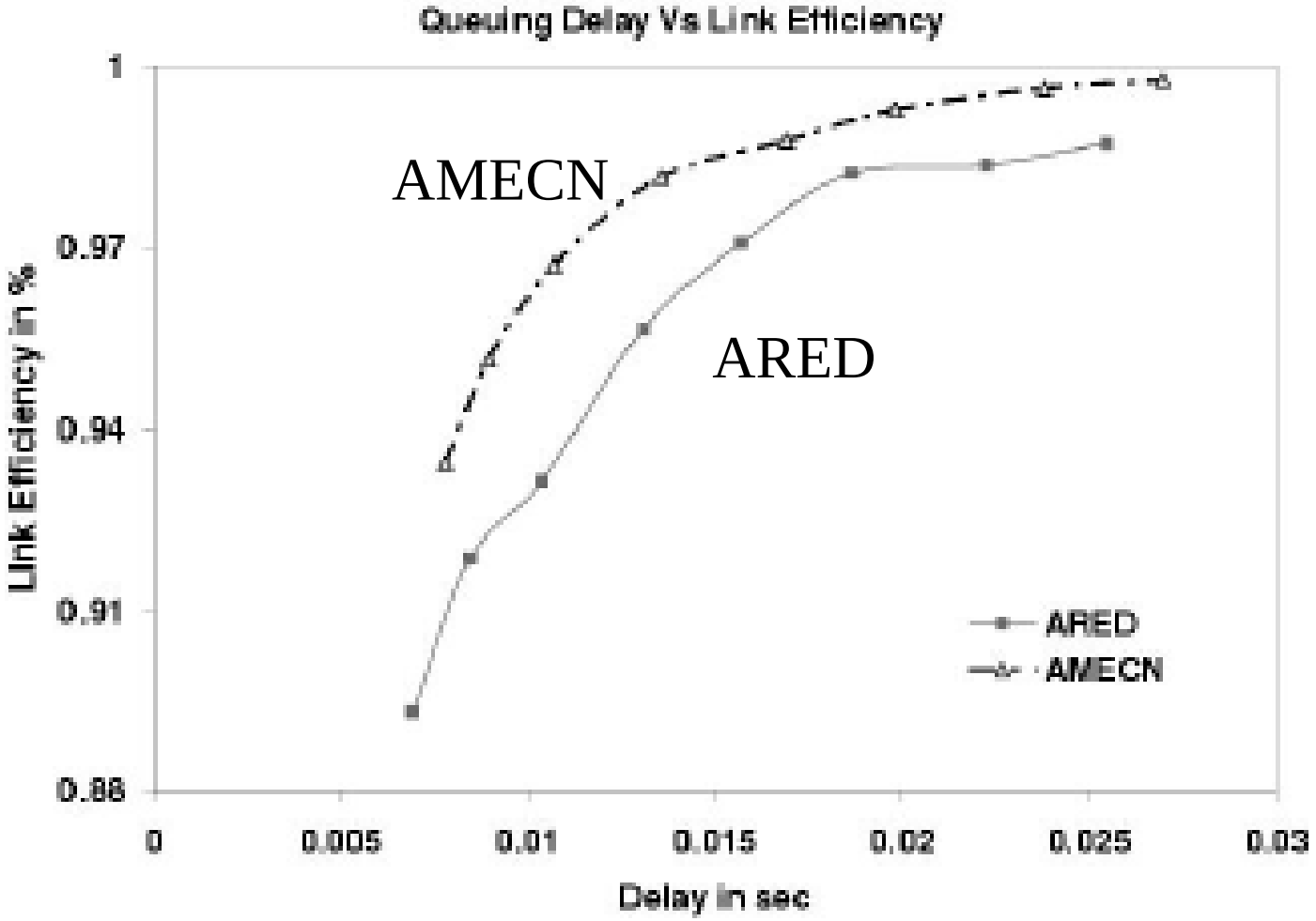
β: decrease factor; $1 - X * \frac{target-ave}{target}$

X: scaling factor; $0.17 * \frac{target}{target-min}$

Simulation Configuration

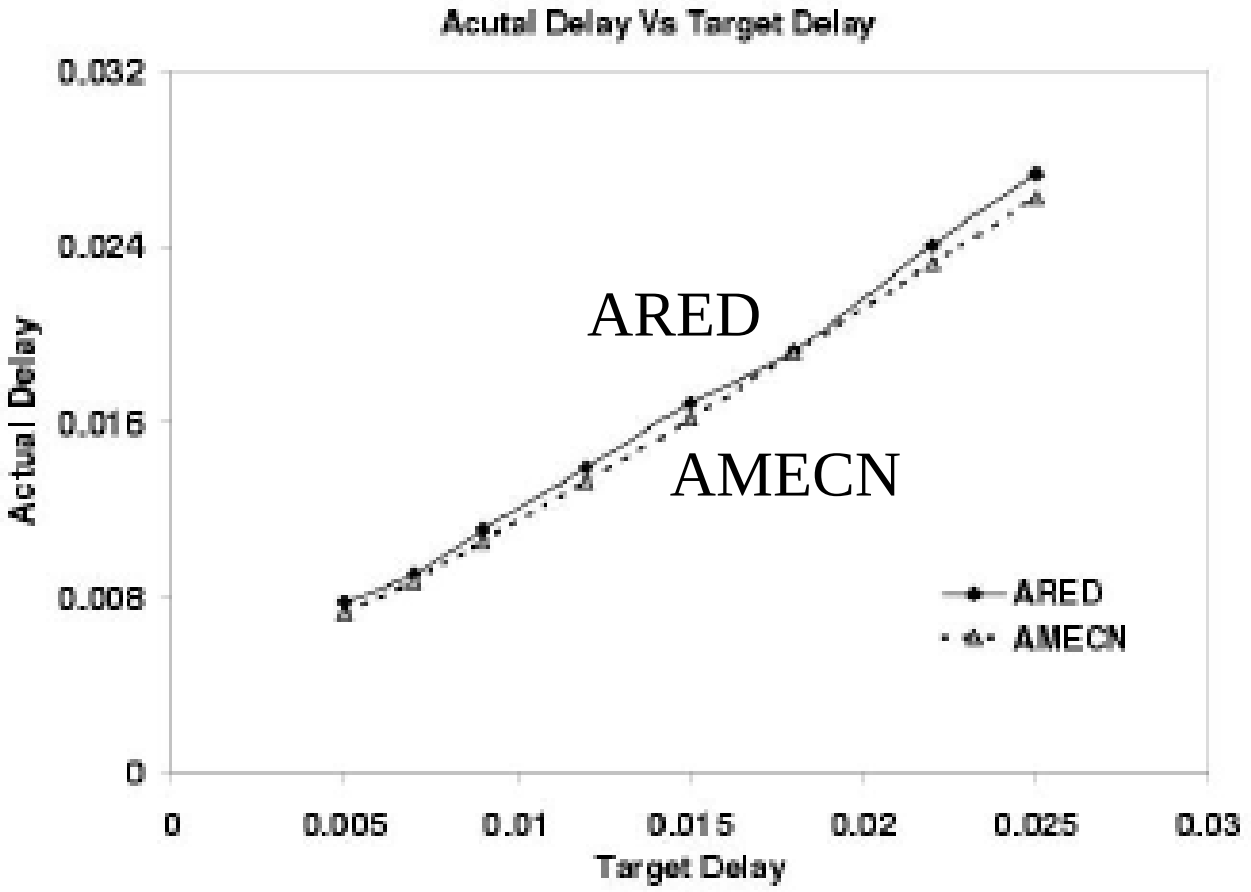


AMECN vs. ARED



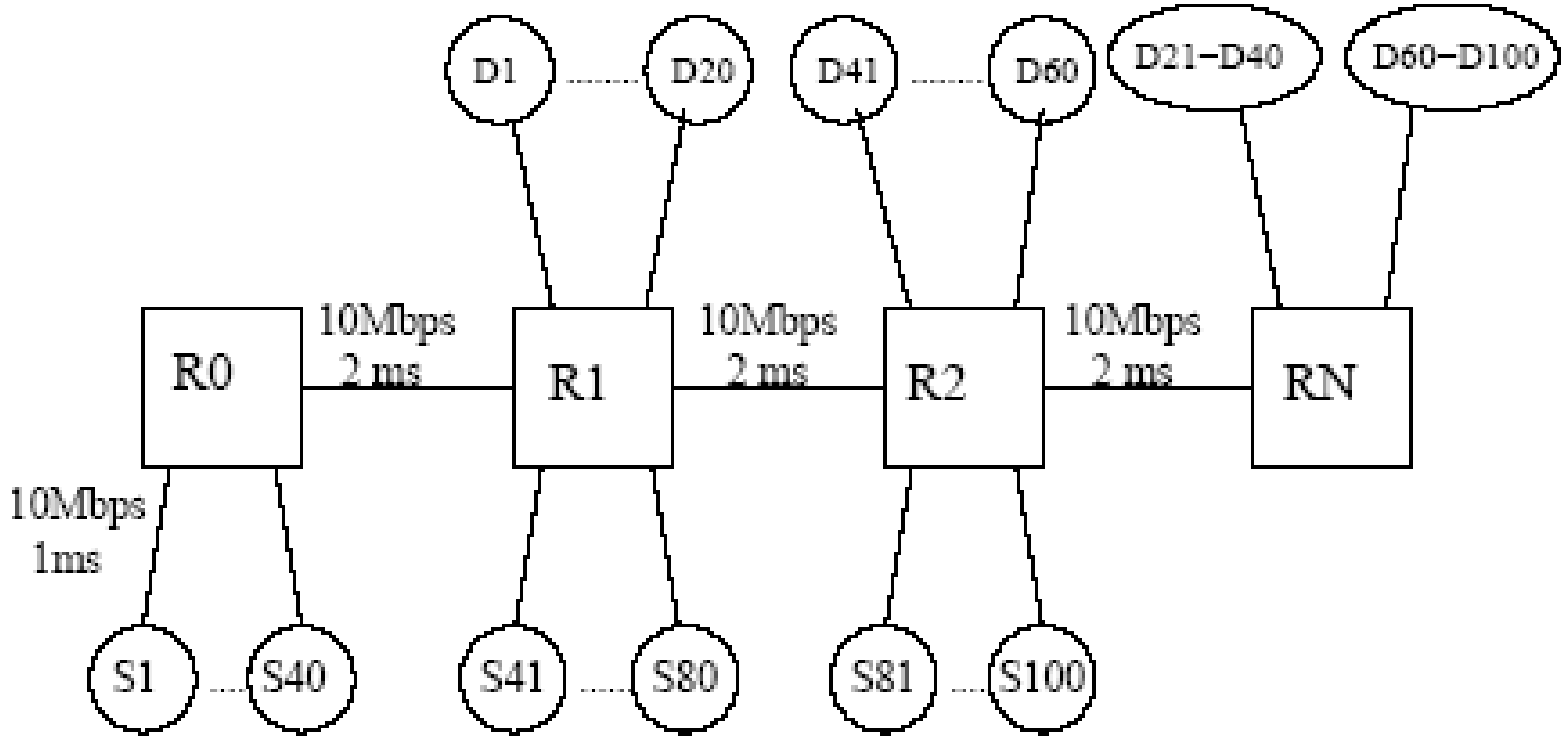
AMECN has a higher link efficiency than ARED.

AMECN vs. ARED

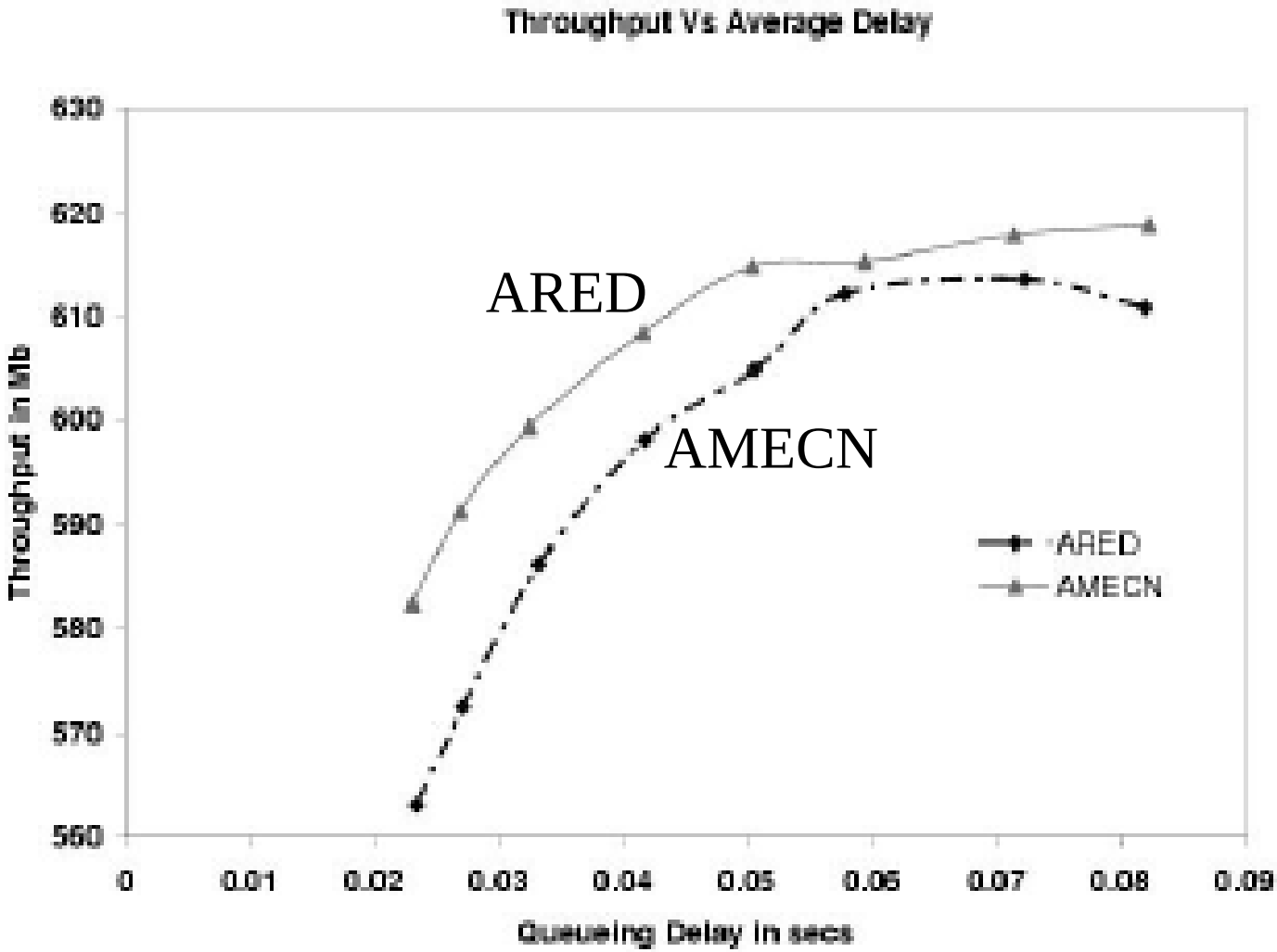


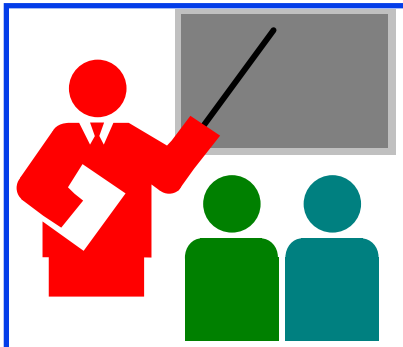
Both AMECN and ARED keep the actual delay close to target delay, but AMECN permits more throughput than ARED

Simulation Configuration for Multiple Congested Gateways



AMECN vs. ARED





Summary

- ECN allows better network efficiency by avoiding packet drops
- Multi-level ECN enhances ECN by allowing multiple queue thresholds
- Adaptive Multilevel ECN enhances MECN by dynamically adopting the maximum probability of marking
- AMECN has better performance than Adaptive RED
  - For same delay – more throughput